THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 61, No. 6, July–August 1982
Printed in U.S.A.

An Improved Training Procedure for Connected-Digit Recognition

By L. R. RABINER, A. BERGH, and J. G. WILPON

The "conventional" way of obtaining word reference patterns for connected-word recognition systems is to use isolated-word patterns, and to rely on the dynamics of the matching algorithm to account for the differences in connected speech. Connected-word recognition, based on such an approach, tends to become unreliable (high-error rates) when the talking rate becomes grossly incommensurate with the rate at which the isolated-word training patterns were spoken. To alleviate this problem, an improved training procedure for connected-word (digit) recognition is proposed in which word reference patterns from isolated occurrences of the vocabulary words are combined with word reference patterns extracted from within connected-word strings to give a robust, reliable word recognizer over all normal speaking rates. The manner in which the embedded-word patterns are extracted was carefully studied, and it is shown that the robust training procedure of Rabiner and Wilpon can be used to give reliable patterns for the embedded, as well as the isolated, patterns. In a test of the system (as a speaker-trained, connected-digit recognizer) with 18 talkers, each speaking 40 different strings (of variable length from 2 to 5 digits), median-string error rates of 0 and 2.5 percent were obtained for deliberately spoken strings and naturally spoken strings, respectively, when the string length was known. Using only isolated-word training tokens, the comparable error rates were 10 and 11.3 percent, respectively.

I. INTRODUCTION

Recently, several algorithms for recognizing a connected string of words based on a set of discrete word reference patterns have been proposed.[1-5] Although the details and the implementations of each of these algorithms differ substantially, basically they all try to find the optimum (smallest distance) concatenation of isolated-word reference patterns that best matches the spoken word string. Thus, the success

of all these algorithms hinges on how well a connected string of words can be matched by concatenating members of a set of isolated-word reference patterns. Clearly, for talking rates that are comparable to the rate of articulation of isolated words [e.g., on the order of 100 words per minute (wpm)], these algorithms have the potential of performing quite accurately.[1–6] However, when the talking rates of the connected-word strings become substantially larger than the articulation rate for isolated words, for example, around 150 wpm, then all the pattern-matching algorithms tend to become unreliable (yield high-error rates). The breakdown mechanism is easily understood in such cases. The first effect of a higher talking rate is a shortening of the duration of the word within the connected-word string. This shortening is highly nonlinear (i.e., vowels and steady fricatives are first shortened and then they are reduced, whereas sound transitions are substantially unaffected) and is not easily compensated by the dynamic time warping (DTW) alignment procedure, which has its own inherent local and global warping path constraints. The second pronounced effect is the sound coarticulation that occurs at the boundaries between words in the string. As the talking rate increases, the amount of coarticulation increases to the point where the isolated-word reference patterns no longer adequately match the words in the sequence, especially at the beginning and end of words.

The above discussion emphasizes the fact that the currently available connected-word recognition systems are inadequate for most reasonable talking rates; i.e., only a highly motivated talker who spoke the connected-word strings in a careful, deliberate manner, would achieve high accuracy. The next question then is what can be done to increase the reliability of the connected-word recognizer at normal talking rates? The answer to this question is the topic of this paper. The proposed solution is an improved training procedure in which the isolated-word reference patterns are supplemented with word reference patterns extracted from connected-word strings. Although there has been no publication on the use of embedded word training patterns for connected-word recognition, the Nippon Electric Company's DP-100 training procedure has, in demonstrations, used embedded digits, obtained as the middle digit of three-digit strings, to supplement the isolated digits in the training set.[7,8]

The purpose of this paper is to show how word reference patterns for a connected-word recognizer can be obtained in a reliable and robust manner from connected-word strings. We also show how such embedded training patterns can be combined with standard isolated-word reference patterns to provide a training set which is capable of recognizing connected-word strings spoken at normal talking rates. In particular, we consider recognition of connected-digit strings of from

two to five digits. It is shown that in a speaker-trained test, the connected-digit recognizer achieved a 97.5 percent median-string accuracy on normally spoken digit strings using the improved training set, whereas an 88.7 percent median-string accuracy was obtained using only isolated-digit templates.

The outline of this paper is as follows. In Section II, we describe the improved training procedure and discuss the important issue of how to extract an embedded-word reference pattern that is reliable and robust. In Section III, we describe and give the results of an experiment to independently evaluate the automatic training procedure of Section II. In Section IV, we present results of a connected-digit recognition test using 18 talkers, each of whom used the improved training procedure. Finally, in Section V, we summarize our results, and discuss alternative procedures for improving the connected-digit recognition algorithm.

II. THE IMPROVED CONNECTED-WORD TRAINING PROCEDURE

To understand the improved connected-word training procedure, we must first review briefly our current implementation of the connected-word recognizer. Thus, we will focus on the level-building (LB) algorithm of Myers and Rabiner,[5,6,9] as this is the algorithm for which all results will be given.

2.1 *Connected-word recognition using a LB approach*

Assume we are given a test string $\mathbf{T} = \mathbf{T}(m)$, $m = 1, 2, \cdots, M$, where $\mathbf{T}$ represents a string of words of unknown length. The connected-word-recognition problem consists of finding the sequence of reference patterns, $\mathbf{R}^s = \mathbf{R}_{q(1)} \circledast \mathbf{R}_{q(2)} \circledast \cdots \circledast \mathbf{R}_{q(L)}$ of arbitrary length L which best matches (minimum distance) the test string $\mathbf{T}$. (The operation $\circledast$ is a concatenation operator). The sequence $\mathbf{R}^s$ defines the L reference patterns, $\mathbf{R}_{q(1)}, \mathbf{R}_{q(2)}, \ldots, \mathbf{R}_{q(L)}$, which, when concatenated, best matches (over all lengths L and over all possible reference patterns) the unknown test sequence $\mathbf{T}$.

The basic procedure for finding $\mathbf{R}^s$ is to solve a time-alignment problem between $\mathbf{T}$ and $\mathbf{R}^s$ using DTW methods. The LB algorithm is basically an efficient procedure for solving the associated DTW minimization.[5,6] Although the details of the exact procedure need not be reviewed here, there are several important properties of the LB algorithm that will be used in this paper. Therefore, we first enumerate these properties, which include:

(*i*) The LB algorithm finds the optimum matching string of every length, L words, from $L = 1$ to $L = \text{LMAX}$ (as set by the user).

(*ii*) For each string match found by the LB algorithm, a segmentation of the test string into appropriate matching regions for each

word in the string is provided. (We will rely on this property to formulate the embedded training procedure).

(*iii*) For every string length, L words, a list of the best Q strings (i.e., the Q lowest distance L-word strings) can be obtained.

(*iv*) By using the flexible parameters of the LB algorithm, namely δ_{R_1} (a range of frames at the beginning of a reference pattern which can be skipped), δ_{R_2} (a range of frames at the end of a reference pattern which can be skipped), and δ_{END}, (a range of frames at the end of the test that can be skipped), modifications of the reference patterns, or, equivalently, large discrete jumps in the warping path, can be made to partially account for word-boundary and word-shortening effects.

Figures 1 and 2 summarize the above properties in a fairly concise manner. Figure 1 shows a sequence of DTW paths corresponding to the best L-word sequence, the best L-1 word sequence, etc. For the best L-word sequence, the segmentation points corresponding to the bound-

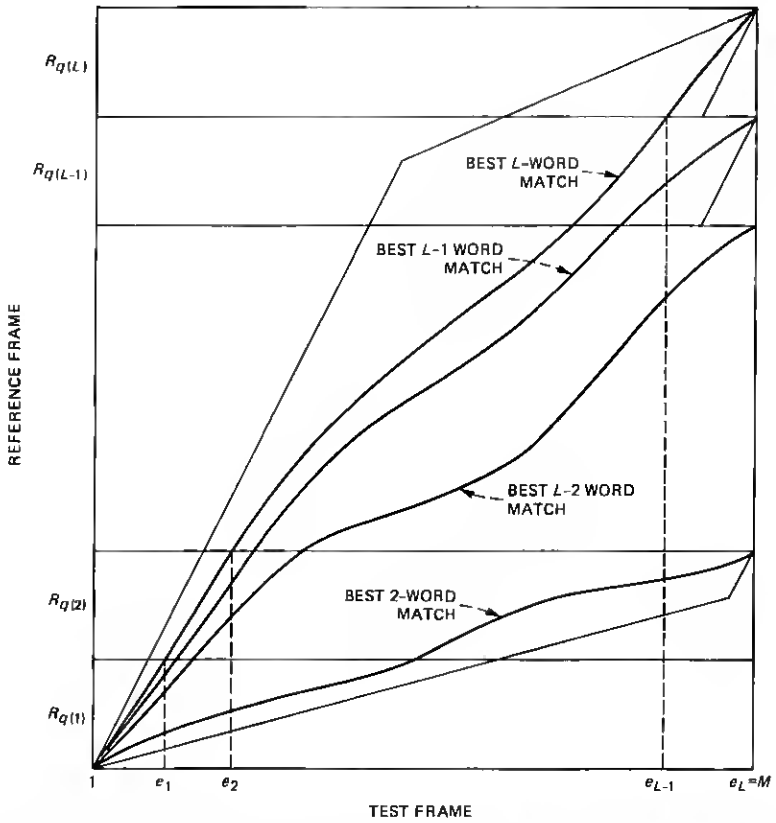

Fig. 1—Sequence of LB DTW warps to provide best word sequences of several different lengths.

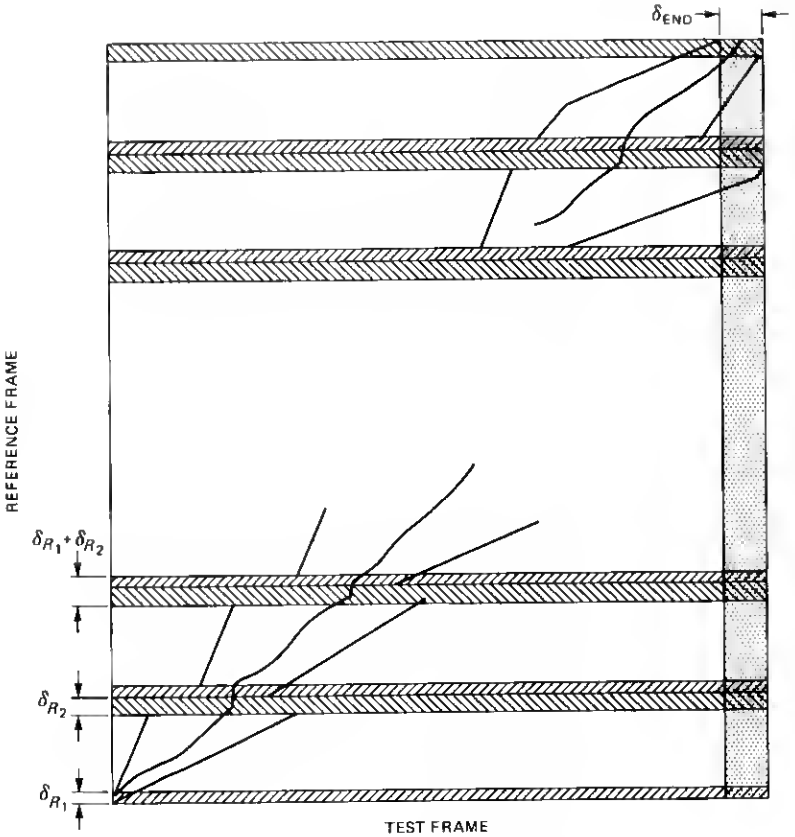

Fig. 2—Summary of flexible LB parameters.

aries of each of the L words in the test string are also shown. It should be clear that the best string of length q need not (and generally won't) contain the same words as the best string of length $q + 1$.

Figure 2 shows a summary of how the LB parameters δ_{R_1}, δ_{R_2}, and δ_{END} can affect the warping path. It shows that the DTW path can literally jump over up to $(\delta_{R_2} + \delta_{R_1})$ frames of the pattern $\mathbf{R}^s$ at each discrete word boundary within $\mathbf{R}^s$. In this manner, a certain degree of word coarticulation and word shortening can be directly accounted for in obtaining the best matches to the test pattern. It has been shown previously[6,7] that judicious use of the LB parameters greatly improves the performance of the LB algorithm for several connected-word recognition tasks.

2.2 *Isolated-word training for connected-word recognition*

The "standard" reference set for the LB algorithm for speaker-trained use is the set of isolated-word patterns obtained via the robust

training procedure of Rabiner and Wilpon.[10] Thus, for each word in the vocabulary, a single-reference token was obtained as the time-warped average of two versions of the word which was deemed sufficiently similar (based on the recognizer-distance measure).

As discussed previously, the major disadvantage of isolated-word reference patterns is that when the talking rate of the test strings becomes much higher than the talking rate of the isolated-word patterns, reliability and accuracy of the connected-word recognizer fall dramatically. To alleviate this problem, we now describe the embedded-word training procedure.

2.3 Embedded-word training for connected-word recognition

The basic idea behind the embedded-word training procedure is that an "improved" set of word reference patterns could be obtained by combining the isolated-word reference patterns with word reference patterns extracted from actual connected-word strings. A second important and related problem is deciding from which strings (i.e., which sequence of words) such patterns should be extracted. Finally, a third related problem is controlling the rate at which the talker speaks the strings from which the embedded words are extracted.

Before describing the training procedure that was studied, it is worthwhile discussing the three points made above. The first point is perhaps the most fundamental one. In connected-word sequences, it is often difficult, if not impossible, to assign boundaries to words in the string. For example, in the sequence/one-nine/, the nasal boundary between the one and the nine is a shared boundary and can arbitrarily be assigned to either digit. As the rate of talking goes up, this problem generally is compounded.

The second problem in implementing an embedded-word training procedure, namely, the choice of strings from which to extract the patterns, involves deciding whether or not to use strings with heavy word coarticulation. (Clearly, this problem is intimately related to the first problem.) For example, the string /616/ will provide a markedly different embedded digit one from the string /219/. Which of these sequences would provide the most useful output, in terms of providing an improved reference pattern for the digit 1, is unclear, and is a topic of investigation in Section III.

The third problem, namely, the talking rate of the talker, again determines the difficulty in extracting the embedded-reference pattern. We will see later in this section that it is reasonable to consider both deliberate (i.e., careful articulation of words) and natural (i.e., normal articulation of words) talking rates for extraction of reference words, and that different analysis procedures should be applied in both cases.

Based on the above discussion, a block diagram of the embedded-

reference word training procedure is given in Fig. 3. The philosophy of the training procedure is similar to that of the robust training procedure for isolated words.[10] For each word in the vocabulary, a sequence generator (i.e., a talker) produces a string of words in which the desired word appears. The choice of word sequences will be discussed below. A DTW alignment procedure matches a set of concatenated word references, corresponding to the words in the string, to the test string T, thereby providing a segmentation of T. The pattern for the appropriate reference word is extracted from T, based on the segmentation, and stored in a temporary word store. It is then compared to all previous occurrences of that word in the store, using another DTW alignment procedure. For each such comparison, a distance score is obtained. If any distance score falls below a specified threshold, then the pair of tokens giving the minimum distance among all versions in the store are averaged, after time alignment, and the resulting reference pattern is saved in an embedded-word store. This procedure is iterated until an embedded pattern is obtained for each word in the vocabulary.

The only unspecified aspect of the embedded-training procedure is the word sequence generator. Since we have tested the improved training procedure only on digit sequences, we will focus our attention solely on this vocabulary. We have considered two sequence generators—a noncoarticulated (NC) sequence generator, and a coarticulated (CO) sequence generator. For both cases, the desired digit was the middle digit of a 3-digit sequence. The NC sequences had the property that, at the boundary, the preceding and following digits had different manners of production than the middle digit. Similarly, the CO sequences had tbe property that either the preceding or the following

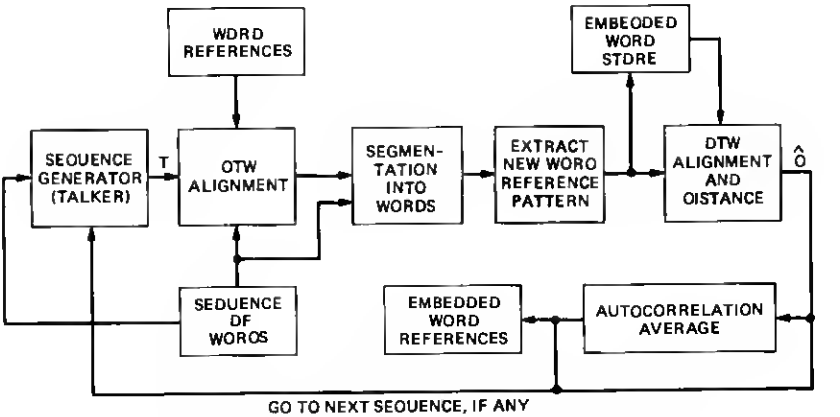

Fig. 3—Block diagram of embedded-digit training procedure.

Table I—Training sequences for embedded digits

NC Sequences			CO Sequences		
614	615	616	919	119	019
123	725	327	020	729	421
234	436	633	438	638	738
346	343	645	341	648	349
256	253	155	651	458	251
261	168	569	866	663	766
173	475	577	671	678	672
668	468	768	668	468	768
693	695	697	191	991	198
104	106	103	601	708	009
681	688	689	388	387	389
617	613	617	011	918	118
624	526	426	921	628	529
335	537	737	531	831	839
247	544	746	549	741	548
357	454	655	759	350	069
369	461	768	667	664	867
974	376	274	679	670	776
568	368	168	568	368	168
694	696	693	199	999	998
903	907	906	309	101	408
689	681	688	386	381	383

digit, or both, had a similar manner of production as the middle digit. Table I shows the actual training sequences for both the NC and CO cases. A total of 66 sequences* were defined for each case representing different environments for each of the digits in strings of a given type.

III. EVALUATION OF THE EMBEDDED-WORD TRAINING PROCEDURE

To study the effectiveness of the embedded-word training procedure, four talkers (two male, two female) each trained a connected-digit recognizer in the following manner:

(*i*) A set of isolated-digit templates was created for each talker using the robust training procedure of Rabiner and Wilpon.[10] A single template was created for each of the ten digits. An eleventh template was created for the digit 8 in which the talker was requested to speak the digit 8 without releasing the final /*t*/. For the normal 8 template, each talker was told to release tbe final /*t*/. We designate the isolated-digit training set as IS.

(*ii*) Four sets of embedded templates were created for each talker, using the robust training procedure of Fig. 3. The four sets of templates differed in speed at which the talker spoke the three-digit training sequences (normal—NR, and deliberate—DL), and the degree of digit coarticulation (coarticulated—CO, and noncoarticulated—NC). Thus,

* For the digit 8, we considered both medial sequences (in which no release of the final *t* was made) and final sequences (in which the final *t* was asked to be released). For final sequences the digit 8 was extracted as the third digit in the string.

the four embedded training sets were denoted as CO.DL, CO.NR, NC.DL, and NC.NR.

The composite reference sets used for testing were created by considering various combinations of the isolated-digit set with one or more of the embedded-digit sets.

Two test sets (TS) of testing data were obtained. The first set, denoted as TS.DL, consisted of 40 randomly chosen (i.e., a different set of 40 strings for each talker) digit strings of varying length from two to five digits, each string spoken deliberately, that is, carefully articulated. The second set, denoted as TS.NR, consisted of the same 40-digit strings for each talker as in TS.DL but, instead, spoken at a normal rate.

Before presenting the results of the evaluation tests, one point is worth noting. Because of the difficulty in accurately finding digit boundaries of a digit embedded in a string of digits, the accuracy of the entire training procedure is suspect since the test recognition scores could be poor because of unreliable embedded-digit boundaries. To check this premise, the test strings used in the embedded-digit training procedure were processed manually, as well as by the automatic algorithm. The manual processing consisted of examining energy plots of the 3-digit string and listening to synthetic versions of the processed digits. This processing was performed iteratively on each string until the experimenter found the most acceptable boundaries for each digit. At this point, the automatic processing of the training procedure took over the rest of the template creation process.

A simple check on the accuracy of the automatic-word extraction process was made by comparing the word boundary frames of the automatic procedure with those of the manual procedure. Histograms were made for each talker for both the initial and final frames of the embedded digit. Typically, the frame error was less than one frame in about two-thirds of the cases, with somewhat larger errors for the remaining one-third of the cases. Figure 4 shows examples of such histograms for one talker. Shown in this figure are the histograms for the initial and final frame errors for two of the embedded-training sets.

A further check on the accuracy of the fully automatic technique was obtained by processing, for recognition, both the automatically obtained embedded-digit reference sets, and the manually obtained embedded-digit reference sets. We now discuss the recognition test results.

3.1 Connected-digits test evaluation of the training sets

For each of the two test sets (TS.DL and TS.NR) and for each talker, the following reference sets were used and tested:

1. IS—Isolated digits only, one template per digit, eleven digits.
2. CO.DL—Embedded digits only, obtained from coarticulated strings, deliberately spoken.

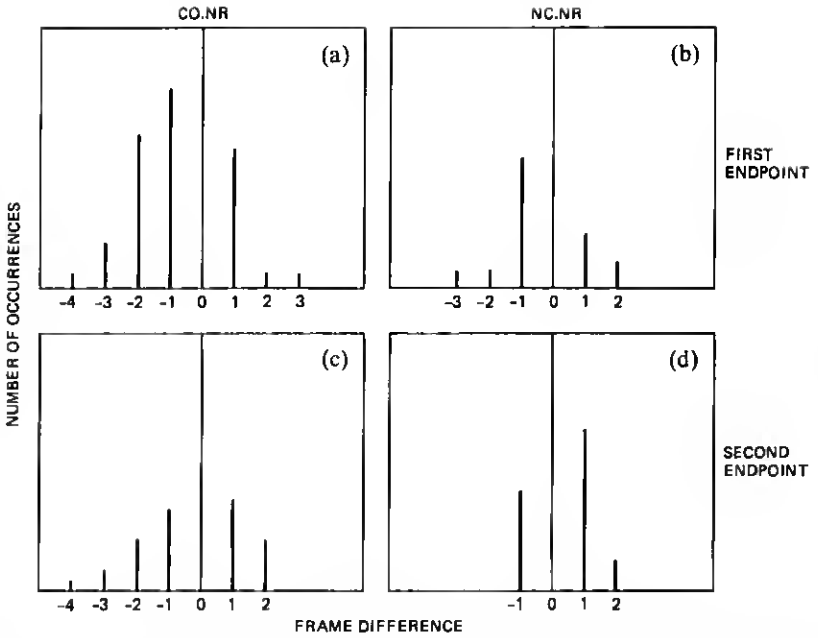

Fig. 4—Histograms of word boundary error (for one of the talkers) between manually and automatically determined boundaries for both (a) and (b) initial and (c) and (d) final boundaries for (a) and (c) CO.NR and (b) and (d) NC.NR.

3. NC.DL—Embedded digits only, obtained from noncoarticulated strings, deliberately spoken.
4. IS ⊕ CO.DL—IS combined with CO.DL.
5. IS ⊕ NC.DL—IS combined with NC.DL.
6. IS ⊕ CO.NR—IS combined with CO.NR.
7. IS ⊕ NC.NR—IS combined with NC.NR.

(Sets 4 to 7 had two templates per digit, eleven digits)

8. IS ⊕ CO.NR ⊕ CO.DL.
9. IS ⊕ CO.NR ⊕ NC.DL.
10. IS ⊕ NC.NR ⊕ CO.DL.
11. IS ⊕ NC.NR ⊕ NC.DL.

(Sets 8 to 11 contained three templates per digit, eleven digits). For reference sets 2 to 11 both automatically and manually obtained templates were used.

Notice that not all single, and double combinations of the embedded sets were used. The sets that were omitted (e.g., CO.NR or NC.NR alone) were found to have high-error rates in preliminary tests; therefore, they were not fully evaluated.

The results of the recognition tests are given in Tables II and III. Table II gives results for TS.DL (the deliberately spoken strings) and

Table II—Recognition results (number of string errors) for different reference sets for TS.DL strings

	Talker Number	Single-Training Sets			Double-Training Sets				Triple-Training Sets			
					IS ⊕				IS ⊕			
		IS	CO.DL	NC.DL	CO.DL	NC.DL	CO.NR	NC.NR	CO.NR ⊕ CO.DL	CO.NR ⊕ NC.DL	NC.NR ⊕ CO.DL	NC.NR ⊕ NC.DL
Automatic endpoints—known-length strings	1	0	5	0	1	0	0	0	1	0	1	0
	2	4	2	2	0	2	2	0	0	1	0	1
	3	5	10	6	4	1	3	3	2	0	2	0
	4	2	0	2	2	2	2	2	2	2	2	2
	Total	11	17	10	7	5	7	5	5	3	5	3
Automatic endpoints—best string	1	0	15	1	2	2	0	1	2	1	3	3
	2	4	9	2	0	2	2	0	1	1	0	1
	3	6	20	10	7	5	9	13	7	6	7	6
	4	9	2	11	8	10	8	9	7	9	8	10
	Total	19	46	34	17	19	19	23	17	17	18	20
Manual endpoints—known-length strings	1	—	2	2	0	0	1	0	0	1	0	0
	2	—	4	1	0	0	0	0	0	0	0	0
	3	—	3	5	2	2	3	2	1	1	1	1
	4	—	0	3	2	2	2	2	2	2	2	2
	Total	—	9	11	4	4	6	4	3	4	3	3
Manual endpoints—best string	1	—	15	9	1	1	3	0	2	1	1	1
	2	—	10	3	3	3	0	0	2	2	2	2
	3	—	11	8	6	5	11	7	5	4	5	4
	4	—	0	11	7	9	8	10	6	8	7	10
	Total	—	36	31	17	18	22	17	15	15	15	17

Table III—Recognition results (number of string errors) for different reference sets for TS.NR strings

		Single-Training Sets			Double-Training Sets				Triple-Training Sets			
					IS ⊕				IS ⊕			
	Talker Number	IS	CO.DL	NC.DL	CO.DL	NC.DL	CO.NR	NC.NR	CO.NR ⊕ CO.DL	CO.NR ⊕ NC.DL	NC.NR ⊕ CO.DL	NC.NR ⊕ NC.DL
Automatic endpoints—known-length strings	1	2	6	2	1	1	0	0	1	0	0	0
	2	10	5	5	1	3	6	2	1	3	1	1
	3	7	3	5	2	2	2	4	3	2	2	1
	4	4	3	5	4	1	3	1	4	1	1	1
	Total	23	17	17	8	7	11	7	9	6	4	3
Automatic endpoints—best string	1	2	13	5	2	1	1	0	2	0	0	0
	2	11	7	5	1	3	6	2	1	3	1	1
	3	11	6	12	4	6	8	9	7	8	5	6
	4	4	3	6	5	2	5	2	6	4	3	3
	Total	28	29	28	12	12	20	13	16	15	9	10
Manual endpoints—known-length strings	1	—	3	2	0	0	1	1	0	0	0	0
	2	—	3	3	1	3	2	2	1	3	1	2
	3	—	4	5	3	2	4	3	2	2	3	1
	4	—	8	6	3	1	2	1	3	1	1	1
	Total	—	18	16	7	6	9	7	6	6	5	4
Manual endpoints—best string	1	—	5	5	1	0	1	1	1	0	0	0
	2	—	7	3	1	3	3	2	1	3	1	2
	3	—	6	7	6	6	12	7	11	10	7	6
	4	—	9	7	3	2	4	2	3	3	1	2
	Total	—	27	22	11	11	20	12	16	16	9	10

Table III is for TS.NR (normally spoken strings). Results are given in this table as the number of string errors (out of a possible 40 strings) when evaluated using the LB DTW string recognizer. The results are given for the best string, regardless of actual string length, and for the best string with string length assumed to be known (KL) a priori. Scores are given for each talker, and a sum of total errors is given at the bottom of each table. A comparison is also provided, when possible, between manually (M) and automatically (A) embedded template sets.

The results of Tables II and III show conclusively that neither isolated- nor embedded-word templates, alone, are adequate for reliable recognition of connected-digit strings spoken at either deliberate or normal rates. It can be seen that the best overall scores were obtained by the triple combination sets with some small variations among them depending on the test set, and the talker. The double combinations provided scores almost as good as the triple combinations.

For the question as to whether to use coarticulated or noncoarticulated sequences in the training, the results of Tables II and III are essentially ambiguous. The scores on the double and triple combinations favor neither type of sequence over the other for both test sets. Similarly, the question as to whether to use deliberately or normally spoken strings in the training set is also left unresolved by the data in Tables II and III. In the next section, we investigate these issues further in a larger test of the overall procedure.

Perhaps the most significant conclusion from Tables II and III is that the error scores from the automatically extracted digit templates are essentially identical (to within small errors) to those of the manually extracted digit templates for all reference sets giving good recognition scores. This result indicated that the automatic training procedure was a useful one for obtaining embedded-digit templates for connected-digit recognition systems.

IV. LARGE-SCALE EVALUATION OF THE IMPROVED TRAINING PROCEDURE

The results of the preliminary test indicated that the improved training procedure led to a reliable set of embedded-digit templates, and that when combined with the normal set of isolated-digit templates, connected-digit, string-recognition scores greatly improved at normal speaking rates over those obtained using isolated-digit templates alone.

To provide a better understanding of the limitations of the improved training procedure, a larger evaluation test was performed in which 18 talkers (nine male, nine female) each used the improved training procedure to provide digit reference patterns. Only two sets of embed-

ded patterns were extracted, namely, NC.DL and CO.NR. The training sets studied in the evaluation included:

(*i*) All three single trainings sets, namely, IS, NC.DL, and CO.NR, each of which consisted of 11 patterns (including the unreleased eight).

(*ii*) The two combinations of isolated and embedded references, namely, IS ⊕ NC.DL and IS ⊕ CO.NR. These sets consisted of 22 patterns.

(*iii*) The triple combination of IS ⊕ NC.DL ⊕ CO.NR. This set consisted of 33 patterns.

As in the previous test, each talker provided two test sets of 40 randomly selected connected-digit strings of from two to five digits. One set was spoken deliberately (TS.DL), the other set was spoken at a normal rate (TS.NR). The digit strings in both sets for the same talker were identical. Different talkers spoke different randomly chosen sets of digit strings.

The recognition test consisted of using the LB-based DTW algorithm to match the spoken test strings by the best sequence of reference patterns, regardless of length. In performing the matches, the parameters of the LB algorithm were set as follows:

(*i*) For all references and test sets, the parameters δ_{END}, M_T, and ϵ were given values of $\delta_{\text{END}} = 4$, $M_T = 1.6$, and $\epsilon = 20$.*

(*ii*) The parameters δ_{R_1} and δ_{R_2} were made to vary with each subset of reference patterns in the following manner. For isolated-digit patterns in all reference sets, the values $\delta_{R_1} = 4$ and $\delta_{R_2} = 6$ were used. For NC.DL reference patterns, the values $\delta_{R_1} = 2$ and $\delta_{R_2} = 3$ were used. For CO.NR reference patterns, the values $\delta_{R_1} = \delta_{R_2} = 0$ were used.

The logic for this choice was that the NC.DL patterns could be shortened somewhat, but not as much as the isolated patterns. However, the CO.NR patterns could not be shortened at all since they came from highly reduced, normally spoken digit sequences.

Before presenting results of the recognition tests, we first give some statistics on each talker's rate of talking.

4.1 Talking rate statistics

The statistics on average talking rate for each of the 18 talkers for both TS.DL and TS.NR strings are given in Table IV, and a summary of the overall average rate (as a function of number of digits in the string) is given in Fig. 5. The talking rates are given in terms of wpm. It can be seen that, for deliberately spoken strings, the average talker's rate varies from 99 to 156 wpm (across the 18 talkers). Thus, a high degree of rate variability exists across talkers for deliberately spoken strings of digits. For naturally spoken digit strings, the average talking rates

* These parameter values were chosen based on preliminary runs on connected-digit strings, as reported in Ref. 6.

Table IV—Average talking rates (wpm) for strings of varying length for TS.DL and TS.NR data

	TS.DL				TS.NR			
	Number of Digits in String				Number of Digits in String			
Talker	2	3	4	5	2	3	4	5
1	118	140	138	143	149	182	176	180
2	125	128	123	128	160	175	160	165
3	124	137	141	142	139	176	179	163
4	108	125	132	127	127	147	154	153
5	124	115	117	121	129	134	141	144
6	106	117	116	117	126	144	140	144
7	121	118	117	111	136	150	159	139
8	125	132	137	139	139	151	155	171
9	130	138	156	148	140	156	169	169
10	105	111	121	122	118	138	135	140
11	147	149	149	142	181	181	182	179
12	107	112	114	123	121	137	147	154
13	110	117	123	120	135	147	153	150
14	121	123	128	129	160	178	181	189
15	123	123	114	119	154	193	171	174
16	99	106	103	110	159	181	183	195
17	151	139	144	137	162	156	162	154
18	107	114	118	128	120	131	158	157
Average	126	131	134	135	150	166	170	171

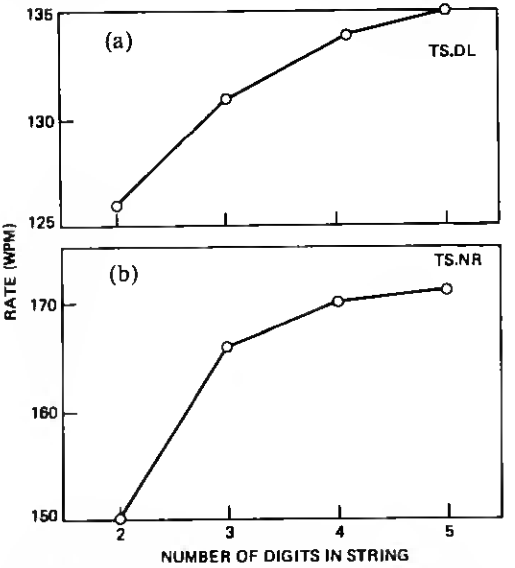

Fig. 5—Plot of average talking rate as a function of the number of digits in the string for (a) TS.DL and (b) TS.NR data.

vary from 118 to 193 wpm (across the 18 talkers), again pointing out the high degree of variability in normal talking rates.

However, the plots in Fig. 5 show that when averaged across talkers, the talking rate for different length strings does not vary as markedly

as for different talkers. For deliberately spoken digit strings, the slowest average rate is 126 wpm for two-digit strings, and the average rate increases to 134 wpm for four-digit strings. Almost no increase in average talking rate is found for five-digit strings over that for four-digit strings. For normally spoken digit strings, the same trends of the average talking rate are seen across different length strings. Thus, the average rate for two-digit strings is 150 wpm and it increases to 170 wpm for four-digit strings. However, the average rate for five-digit normally spoken strings (171 wpm) is essentially the same as for four-digit normally spoken strings.

4.2 Recogniton test results on large data base

The results of the recognition tests on the 18-talker data base are given in Tables V and VI, and are plotted in Fig. 6. Tables V and VI give average- and median-string error rates (averaged over talkers and

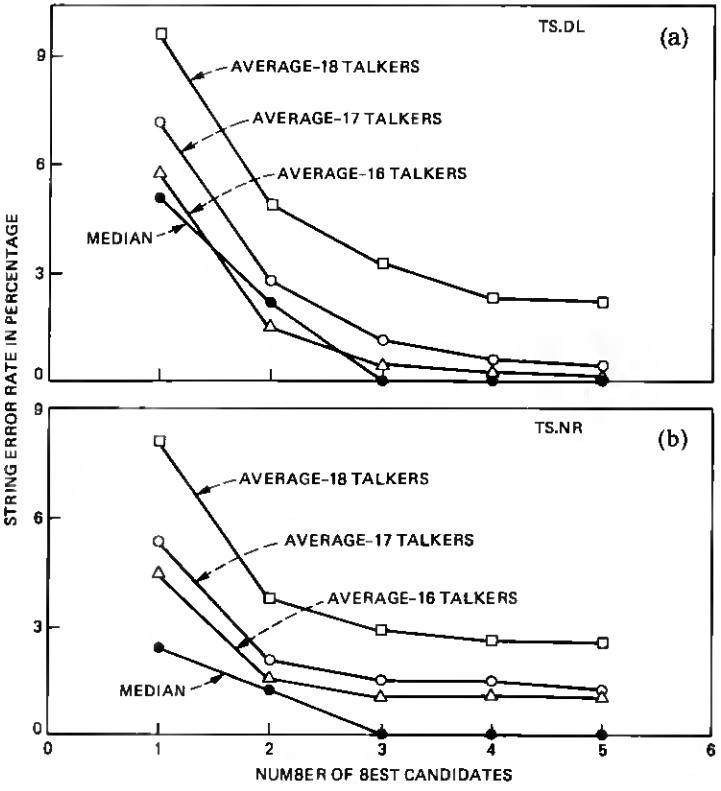

Fig. 6—Plots of string error rates versus number of best candidates for (a) TS.DL and (b) TS.NR data using reference set IS ⊕ NC.DL ⊕ CO.NR.

Table V—Average- and median-string error rates (percentage errors) for TS.DL data for the three reference sets

		TS.DL					
		Number of Best Candidates					
Reference Set	Statistic	KL	1	2	3	4	5
IS	Median	10.0	16.3	8.8	6.3	5.0	5.0
	18 Talkers	11.4	20.0	11.8	9.9	8.5	8.3
	17 Talkers	9.1	17.6	9.3	7.8	6.6	6.5
	16 Talkers	7.5	16.3	8.3	6.4	5.5	5.3
IS ⊕ NC.DL	Median	2.5	10.0	2.5	2.5	2.5	2.5
	18 Talkers	5.7	14.1	7.4	5.3	5.0	4.9
	17 Talkers	3.1	12.2	4.9	3.1	2.9	2.8
	16 Talkers	2.0	10.5	3.4	2.0	1.9	1.9
IS ⊕ NC.DL ⊕ CO.NR	Median	0	5.0	2.5	0	0	0
	18 Talkers	3.3	9.7	4.9	3.3	2.6	2.5
	17 Talkers	1.3	7.2	2.8	1.2	0.6	0.4
	16 Talkers	1.1	5.8	1.9	0.5	0.3	0.2

Table VI—Average- and median-string error rates (percentage errors) for TS.NR data for the three reference sets

		TS.NR					
		Number of Best Candidates					
Reference Set	Statistic	KL	1	2	3	4	5
IS	Median	11.3	13.8	10.0	7.5	7.5	7.5
	18 Talkers	16.3	19.0	11.7	10.3	10.1	9.7
	17 Talkers	13.1	15.9	9.3	8.1	7.9	7.5
	16 Talkers	13.0	15.5	9.1	8.0	7.8	7.3
IS ⊕ NC.DL	Median	5.0	7.5	3.8	3.8	3.8	3.8
	18 Talkers	8.3	11.5	6.4	5.8	5.4	5.3
	17 Talkers	5.3	8.4	4.0	3.7	3.2	3.1
	16 Talkers	4.8	7.8	3.3	3.1	2.7	2.7
IS ⊕ NC.DL ⊕ CO.NR	Median	2.5	2.5	1.3	0	0	0
	18 Talkers	5.8	8.1	3.8	2.9	2.6	2.5
	17 Talkers	3.7	5.4	2.1	1.5	1.5	1.3
	16 Talkers	3.4	4.5	1.6	1.1	1.1	1.1

various length strings) for three of the reference sets studied* (IS, IS ⊕ NC.DL, IS ⊕ NC.DL ⊕ CO.NR) for TS.DL data (Table V) and TS.NR data (Table VI). Included in the table are string error rates based on the top 1, 2, 3, 4, and 5 candidates (regardless of string length) and for the case in which the string length was known a priori (KL); that is, we only considered strings of the proper length. Results are given for the median error rate, the average error rate for all 18 talkers, the average error rate for 17 of the 18 talkers (omitting the one with the highest

* These three reference sets were chosen as they gave the best recognition rates and were representative of previously used training methods.

error rate), and the average error rate for 16 of the 18 talkers (omitting the two talkers with the highest error rates).

Figure 6 shows plots of the average and median error rates as a function of the top n candidates (n = 1, 2, 3, 4, 5) for both TS.DL and TS.NR data using the reference set IS ⊕ NC.DL ⊕ CO.NR, which provided the best overall results.

The results of Tables IV and V and Fig. 6 show the following:

(*i*) For deliberately spoken strings (TS.DL), the median-string error rate is 5 percent on the top candidate and falls to 2.5 percent on the top two candidates for reference set IS ⊕ NC.DL ⊕ CO.NR. Using reference set IS alone, the median error rate is 16.3 percent on the top candidate and is still 8.8 percent on the top two candidates. Using reference set IS ⊕ NC.DL, the median error rate is 10.0 percent on the top candidate and 2.5 percent on the top two candidates.

(*ii*) For deliberately spoken strings, the median error rates with known-length strings are 0 percent for reference set IS ⊕ NC.DL ⊕ CO.NR, 10 percent for reference set IS, and 2.5 percent for reference set IS ⊕ NC.DL.

(*iii*) For deliberately spoken strings, the average error rate scores for all 18 talkers are significantly larger than the median error rate scores. However, when the talker with the highest error rate is omitted, i.e., only 17 talkers are used), the average and median error rate scores are comparable.

(*iv*) For normally spoken strings (TS.NR), the median-string error rate is 2.5 percent on the top candidate and falls to 1.3 percent on the top two candidates using reference set IS ⊕ NC.DL ⊕ CO.NR. Using reference set IS alone, the median error rate is 13.8 percent on the top candidate and 10 percent on the top two candidates. Using reference set IS ⊕ NC.DL, the median error rates are 7.5 percent on the top candidate and 3.8 percent on the top two candidates.

(*v*) For normally spoken strings, the median error rates with known-length strings are 2.5 percent, 11.3 percent, and 5.0 percent, for reference sets IS ⊕ NC.DL ⊕ CO.NR, IS, and IS ⊕ NC.DL, respectively.

(*vi*) For normally spoken strings, the average error rate scores for all 18 talkers are significantly larger than the median error rate scores. Again, when the talker with the highest error rate is omitted (the same one as for deliberate strings), the average and median error rate scores become comparable.

From the above set of results, we can draw the following conclusions:

(*i*) The inclusion of embedded-digit training led to significant improvements in digit-recognition accuracy for both deliberately and normally spoken digit strings.

(*ii*) Somewhat better recognition accuracy was obtained when the string length of the test sequence was known than when no knowledge

of string length was used. This is due to the high probability of inserting extraneous sbort digits (tbe embedded twos and eights) in deliberately, and often naturally, spoken strings.

(*iii*) The accuracy with which connected-digit strings could be recognized can be made essentially independent of the talking rate, especially if one can take advantage of knowing in advance the length of the digit string.

(*iv*) Both coarticulated and noncoarticulated embedded-digit training patterns aid in recognizing connected strings of digits.

(*v*) There are some talkers (1 of the 18 tested here) for whom all the training procedures failed. For this one talker, the string error rate exceeded 50 percent on all reference and test sets. No obvious or clear explanation is available for this result, except for the fact that all the training procedures indicated a high degree of variability in speaking digits for this talker; that is, it took the maximum number of iterations to obtian each template set. This higb degree of variability basically implied that no single set of reference patterns could adequately match the digits of this talker. Hence, highly inaccurate connected-digit recognition resulted.

The above results and the conclusions drawn from them indicated that the improved training procedure provided, in general, robust, reliable digit patterns that greatly aided the connected-digit recognizer in recognizing connected-digit strings for these talkers at essentially any reasonable talking rate.

4.3 Speaker-independent recognition using only isolated-digit training

Although the improved training procedure could be incorporated into a clustering analysis to provide speaker-independent, embedded-digit patterns, such an effort has not yet been undertaken. However, to provide a yardstick for comparison, a speaker-independent reference

Table VII—Median- and average-string error rates for speaker-independent recognition of connected digits

	Number of Best Candidates					
Statistic	KL	1	2	3	4	5
			TS.DL			
Median	5.0	12.5	5.0	2.5	2.5	0
18 Talkers	6.7	14.3	5.3	3.9	3.3	2.4
17 Talkers	6.0	12.9	4.3	3.2	2.7	1.8
16 Talkers	5.3	12.0	3.6	3.0	2.3	1.7
			TS.NR			
Median	12.5	20.0	10.0	5.0	2.5	2.5
18 Talkers	14.2	10.3	10.3	8.1	6.3	5.7
17 Talkers	12.7	16.8	8.7	6.6	5.0	4.4
16 Talkers	10.9	14.8	7.5	5.3	3.9	3.4

set consisting of 12 isolated-digit templates for each digit was used in the LB-based DTW recognizer using TS.DL and TS.NR data. The results of these recognition tests are given in Table VII. It can be seen that for TS.DL, the average- and median-string error rates for the 18-talker population are comparable to those of the speaker-trained sets of Table V. (This result is expected since tbe variability of the 12 patterns of each digit is adequate for representing inherent digit variations in deliberately spoken strings.) However, for TS.NR, the average- and median-string error rates using speaker-independent, isolated-digit references are much larger (on the order of 2:1) than those of Table VI for the speaker-trained case. Tbese results again demonstrate the effectiveness of the embedded-digit training for recognizing digits in normally spoken strings.

V. DISCUSSION

The results given in Section IV show that for speaker-trained, connected-digit recognition, high reliability and accuracy can be obtained across a fairly broad range of talking rates by combining embedded-digit training patterns with the standard isolated-digit training patterns. The question that now remains is how useful is such a procedure for real-world applications, and what general types of problems remain. The answer to the first part of the question seems clear. For talkers who are fairly repeatable in the way in which they speak digits, both in isolation and in sequences, this improved training procedure should make connected-digit recognition a viable procedure. For the remaining talkers (e.g., the talker who had substantial digit variability), this procedure is too simplistic and cannot adequately represent the variations in pronunciation of digits. An analysis of the errors made in trying to recognize the strings of such talkers shows no consistent error pattern; that is, the method basically breaks down because of the high variability in the spoken-digit strings. In a sense, the entire training procedure has broken down for such talkers; therefore, the matching procedure is doomed to failure.

The answer to the second part of the main question as to what general types of problems remain is not at all clear. A detailed examination of the types of errors that occurred in all the test sets indicates a fairly broad mix of digit insertions, digit deletions, and digit substitutions. It would appear that for most of these errors there is no simple fix. In these cases, we are basically at or near the limits at which a matching procedure can operate. In such cases, a feature-based, connected-digit recognizer would appear to offer more promise in that it would be less sensitive to parametric variability so long as the detected features remain in the acoustic signal. Since all spoken strings were highly intelligible to the experimenters, it would seem that the promise

of perfect-digit recognition needs a less-sensitive parameterization, i.e., one more closely related to phonetic features, to be fulfilled.

VI. SUMMARY

An improved training procedure for extracting reference-word templates for connected-word recognition systems has been described. The resulting reference patterns essentially model the word characteristics when embedded in connected-word strings. Hence, a reference set consisting of botb isolated-word patterns and embedded-word patterns has the capability of providing reliable recognition of connected-word strings spoken at natural rates. In an evaluation of this procedure and using a digit vocabulary, it was shown that high-string accuracy could be obtained, if the length of the digit string was known a priori, for deliberately and naturally spoken digit strings. If the length of the digit was not known, the string error rate was somewhat higher due to the problem of inserting short digits into the matching sequence.

REFERENCES

1. T. K. Vintsyuk, "Element-Wise Recognition of Continuous Speech Composed of Words from a Specified Dictionary," Kibernetika, *2* (April 1971), pp. 133–43.
2. J. S. Bridle and M. D. Brown, "Connected Word Recognition Using Whole Word Templates," Proc. Inst. Acoustics, Autumn Conf. (November 1979), pp. 25–8.
3. H. Sakoe, "Two Level DP-matching—A Dynamic Programming Based Pattern Matching Algorithm for Connected Word Recognition," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-27*, No. 6 (December 1979), pp. 588–95.
4. L. R. Rabiner and C. E. Schmidt, "Application of Dynamic Time Warping to Connected Digit Recognition," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-28*, No. 4 (August 1980), pp. 377–88.
5. C. S. Myers and L. R. Rabiner, "A Level Building Dynamic Time Warping Algorithm for Connected Word Recognition," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-29*, No. 2 (April 1981), pp. 284–97.
6. C. S. Myers and L. R. Rabiner, "Connected Digit Recognition Using a Level Building DTW Algorithm," IEEE Trans. Acoustics, Speech, and Signal Processing, *ASSP-29*, No. 3 (June 1981), pp. 351–63.
7. S. Tsuruta et al., "Connected Speech Recognition System DP-100," NEC Res. and Dev., *56* (January 1980), pp. 88–93.
8. S. Tsuruta, "DP-100 Voice Recognition System Achieves High Efficiency," J. Eng. Ed., (July 1978), pp. 50–4.
9. C. S. Myers and L. R. Rabiner, "A Comparative Study of Several Dynamic Time Warping Algorithms for Connected Word Recognition," B.S.T.J. *60*, No. 7 (September 1981), pp. 1389–09.
10. L. R. Rabiner and J. G. Wilpon, "A Simplified, Robust Training Procedure for Speaker Trained, Isolated Word Recognition," J. Acoust. Soc. Am., *68*, No. 5 (November 1980), pp. 1271–6.